

Research Article

Face to Face: Anthropometry-Based Interactive Face Shape Modeling Using Model Priors

Yu Zhang¹ and Edmond C. Prakash²

¹*Institute of High Performance Computing, 1 Fusionopolis Way, 16-16 Connexis, Singapore 138632*

²*Department of Computing and Mathematics, Manchester Metropolitan University, Manchester M1 5GD, UK*

Correspondence should be addressed to Yu Zhang, zhangyu.luo@hotmail.com

Received 1 February 2009; Accepted 19 February 2009

Recommended by Suiping Zhou

This paper presents a new anthropometrics-based method for generating realistic, controllable face models. Our method establishes an intuitive and efficient interface to facilitate procedures for interactive 3D face modeling and editing. It takes 3D face scans as examples in order to exploit the variations presented in the real faces of individuals. The system automatically learns a model prior from the data-sets of example meshes of facial features using principal component analysis (PCA) and uses it to regulate the naturalness of synthesized faces. For each facial feature, we compute a set of anthropometric measurements to parameterize the example meshes into a measurement space. Using PCA coefficients as a compact shape representation, we formulate the face modeling problem in a scattered data interpolation framework which takes the user-specified anthropometric parameters as input. Solving the interpolation problem in a reduced subspace allows us to generate a natural face shape that satisfies the user-specified constraints. At runtime, the new face shape can be generated at an interactive rate. We demonstrate the utility of our method by presenting several applications, including analysis of facial features of subjects in different race groups, facial feature transfer, and adapting face models to a particular population group.

Copyright © 2009 Y. Zhang and E. C. Prakash. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. Introduction

One of the most challenging tasks in graphics modeling is to build an interactive system that allows users to model varied, realistic geometric models of human faces quickly and easily. Applications of such a system range from entertainment to communications: virtual human faces need to be generated for movies, computer games, advertisements, or other virtual environments, and facial avatars are needed for video teleconference and other instant communication programs. Some authoring tools for character modeling and animation are available (e.g., Maya [1], Poser [2], DazStudio [3], PeoplePutty [4]). In these systems, deformation settings are specified manually over the range of possible deformation for hundreds of vertices in order to achieve desired results. An infinite number of deformations exist for a given face mesh that can result in different shapes ranging from the realistic facial geometries to implausible appearances. Consequently, interactive modeling is often a tedious and complex process requiring substantial technical as well as

artistic skill. This problem is compounded by the fact that the slightest deviation from real facial appearance can be immediately perceived as wrong by the most casual viewer. While the existing systems have exquisite control rigs to provide detailed control, these controls are based on general modeling techniques such as point morphing or free-form deformations, and therefore lack intuition and accessibility for novices. Users often face a considerable learning curve to understand and use such control rigs.

To address the lack of intuition in current modeling systems, we aim to leverage the anthropometrical measurements as control rigs for 3D face modeling. Traditionally, anthropometry—the study of human body measurement—characterizes the human face using linear distance measures between anatomical landmarks or circumferences at predefined locations [5]. The anthropometrical parameters provide a familiar interface while still providing a high level of control to users. While this is a compact description, they do not uniquely specify the shape of the human face. Furthermore, particularly for computer face modeling,

the sparse anthropometric measurements taken at a small number of landmarks on the face do not capture the detailed shape variations needed for realism. The desire is to map such sparse data into a fully reconstructed 3D surface model. Our goal is a system that uses model priors learned from prerecorded facial shape data to create natural facial shapes that match anthropometric constraints specified by the user. The system can be used to generate a complete surface mesh given only a succinct specification of the desired shape, and it can be used by expert and novice alike to create synthetic 3D faces for myriad uses.

1.1. Background and Previous Work. A large body of literature on modeling and animating faces has been published in the last three decades. A good overview can be found in the textbook [6] and in the survey [7]. In this work, we focus on modeling static face geometry. In this context, several approaches have been proposed. They can be roughly classified into the creative approach and the reconstructive approach.

The creative approach is to facilitate manual specification of the new face model by a user. Parametric face models [8–11] and many commercial modelers fall into this approach. The desire is to create an encapsulated model that can generate a wide range of faces based on a small set of input parameters. They provide full control over the result, including the ability to produce cartoon effects and the high efficiency of geometric manipulation. However, manual parameter tuning without geometric constraints from real human faces for generating realistic faces is difficult and time-consuming. Moreover, the choice of the parameter set depends on the face mesh topology and therefore the manual association of a group of vertices to a specific parameter is required.

The reconstructive approach is to extract face geometry from the measurement of a living subject. The reconstructive approach is to extract face geometry from the measurement of a living subject. In this category, the image-based technique [12–18] utilizes an existing 3D face model and information from few pictures (or video streams) for the reconstruction of face geometry. Although this kind of technique can provide reconstructed face models easily, its drawbacks are the inaccurate geometry reconstruction and inability to generate new faces that have no image counterparts. Another limiting factor of this technique lies in that it gives very little control to the user.

With a significant increase in the quality and availability of 3D capture methods, a common approach towards creating face models uses laser range scanners to acquire both the face geometry and texture simultaneously [19–22]. Although the acquired face data is highly accurate, unfortunately, substantial effort is needed to process the noisy and incomplete data into a model suitable for modeling or animation. In addition, the result of this effort is a model corresponding to a single individual; and each new face must be found on a subject. The desired face may not even physically exist. Furthermore, the user does not have any control over the captured model to edit it in a way that produces a novel face.

Besides these approaches, DeCarlo et al. [23] construct a range of face models with realistic proportions using a variationally constrained optimization technique. However, without the use of the model priors, their method cannot generate natural models unless the user accurately specifies a very detailed set of constraints. Also, this approach requires minutes of computation for the optimization process to generate a face model. Blanz and Vetter [24] present a process for estimating the shape of a face from a single photograph. This is extended by Blanz et al. [25], who present a set of controls for intuitive manipulation of facial attributes. In contrast to our work, they manually assign attribute values to characterize the face shape, and devise attribute controls using linear regression. Vlastic et al. [26] use multilinear face models to study and synthesize variations in faces along several axes, such as identity and expression. An interface for gradient-based face space navigation has been proposed in [27]. Principal components that are not intuitive to users are used as navigation axes in face space, and facial features cannot be controlled individually. The authors focus on a comparison of different user interfaces.

Several commercial systems for generating composite facial images are available [28–30]. Although they are effective to use, a 2D face composite still lacks some of the advantages of a 3D model, such as the complete freedom of viewpoint and the ability to be combined with other 3D graphics. Additionally, to our knowledge, no commercial 2D composite system available today supports automatic completion of unspecified facial regions according to statistical properties. FaceGen 3 [31] is the only existing system that we have found to be similar to ours in functionality. However, there is not much information available about how this function is achieved. As far as we know, it is built on [24] and the face mesh is not divided into different independent regions for localized deformation. In consequence, editing operations on individual facial features tend to affect the whole face.

1.2. Our Approach. In this paper, we present a new method for interactively generating facial models from user-specified anthropometric parameters while matching the statistical properties of a database of scanned models. Figure 1 shows a block diagram of the system architecture. We use a three-step model fitting approach for the 3D registration problem. By bringing scanned models into full correspondence with each other, the shape variation is represented by using principal component analysis (PCA), which induces a low-dimensional subspace of facial feature shapes. We explore the space of probable facial feature shapes using high-level control parameters. We parameterize the example models using the face anthropometric measurements, and predefine the interpolation functions for the parameterized example models. At runtime, the interpolation functions are evaluated to efficiently generate the appropriate feature shapes by taking the anthropometric parameters as input. Apart from an initial tuning of feature point positions, our method works fully automatically. We evaluate the performance of our method with cross-validation tests. We also compare our method against optimization in the PCA

subspace for generating facial feature shapes from constraints of the ground truth data.

In addition, the anthropometric-based face synthesis method, combined with our database of statistics for a large number of subjects, opens ground for a variety of applications. Chief among these is analysis of facial features of different races. Second, the user can transfer facial feature(s) from one individual to another. This allows a plausible new face to be quickly generated by composing different features from multiple faces in the database. Third, the user can adapt the face model to a particular population group by synthesizing characteristic facial features from extracted statistics. Finally, our method allows for compression of data, enabling us to share statistics with the research community for further study of faces.

Unlike a previous approach [23], we utilize the prior knowledge of the face shape in relation to the given measurements to regulate the naturalness of modeled faces. Moreover, our method efficiently generates a new face with the desired shape within a second. Our method also differs significantly from the approach presented in [24, 25] in several respects. First, they manually assign the attribute values to the face shape and devise attribute controls for single control using linear regression. We automatically compute the anthropometric measurements for face shape and relate several attribute controls simultaneously by learning a mapping between the anthropometric measurement space and the feature shape space through scattered data interpolation. Second, they use a 3D variant of a gradient-based optical flow algorithm to derive the point-to-point correspondence between scanned models. This approach does not work well for faces of different races or in different illumination given the inherent problem of using static textures. We present a robust method of determining correspondences that does not depend on the texture information. Third, their method tends to control the global face and requires additional constraints to restrict the effect of editing operations to a local region. In contrast, our method guarantees local control thanks to its feature-based nature.

The main contributions of our work are as follows.

- (i) A general, controllable, and practical system for face modeling and editing. Our method estimates high-level control models in order to infer a particular face from intuitive input controls. As correlations between control parameters and the face shape are estimated by exploiting the real faces of individuals, our method regulates the naturalness of synthesized faces. Unspecified parts of the synthesized facial features are automatically completed according to statistical properties.
- (ii) We propose a new algorithm which uses intuitive attribute parameters of facial features to navigate face space. Our system provides sets of comprehensive anthropometric parameters to easily control face shape characteristics, taking into account the physical structure of real faces.
- (iii) A robust, automatic model fitting approach for establishing correspondences between scanned models.

- (iv) The automatic runtime synthesis is efficient in time complexity and performs fast.

The remainder of this paper is organized as follows: Section 2 presents the face data we use. Section 3 elaborates on the model fitting technique. Section 4 describes the construction of local shape spaces. The face anthropometric parameters used in our work are illustrated in Section 5. Section 6 and Section 7 describe our techniques of feature-based shape synthesis and subregion blending, respectively. After presenting and explaining the results in Section 8, we present a variety of applications of our approach in Section 9. Section 10 gives concluding remarks and our future work.

2. Scanned Data and Preprocessing

We use the USF face database [32] that consists of Cyberware face scans of 186 subjects with a mixture of gender, race, and age. The age of the subjects ranges from 17 to 68 years, and there are 126 male and 60 female subjects. Most of the subjects are Caucasians (129), with African-Americans making up the second largest group (37), and Asians the smallest group (20). All faces are without makeup and accessories. The laser scans provide face structure data which contains approximately 180 k surface points and a 360×524 reflectance (RGB) image for texture-mapping (see Figures 2(a) and 2(b)). We also use a generic head model which consists of 1.092 vertices and 2.274 triangles. Prescribed colors are added to each triangle to form a smooth-shaded surface (see Figure 2(c)).

Let each 3D face scan in the database be S_i ($i = 1, \dots, M$). Since the number of vertices in S_i varies, we resample all faces in the database so that they have the same number of vertices all in mutual correspondence. Feature points are identified semi-automatically to guide the resampling. Figure 3 depicts the process. As illustrated in Figure 3(a), a 2D *feature mask* consisting of polylines groups a set of 86 feature points that correspond to the feature point sets of MPEG-4 Facial Definition Parameters (FDPs) [33]. The feature mask is superimposed onto the front-view face image obtained by orthographic projection of a textured 3D face scan into an image plane. The facial features in this image are identified by using the Active Shape Models (ASMs) [34] and the feature mask is fitted to the features automatically. The 2D feature mask can be manipulated interactively. A little user interaction is needed to tune the feature point positions due to the slight inaccuracy of the automatic facial feature detection. But this is restricted to slight corrections of wayward feature points. The 3D positions of the feature points on the scanned surface are then recovered by back-projection to the 3D space. In this way, we efficiently define a set of feature points on a scanned model S_i as $U_i = \{\mathbf{u}_{i,1}, \dots, \mathbf{u}_{i,n}\}$, where $n = 86$. Our generic model G is already tagged with the corresponding set of feature points $V = \{\mathbf{v}_1, \dots, \mathbf{v}_n\}$ by default.

3. Model Fitting

3.1. Global Warping. The problem of deriving full correspondence for all models S_i can be stated as: resample the

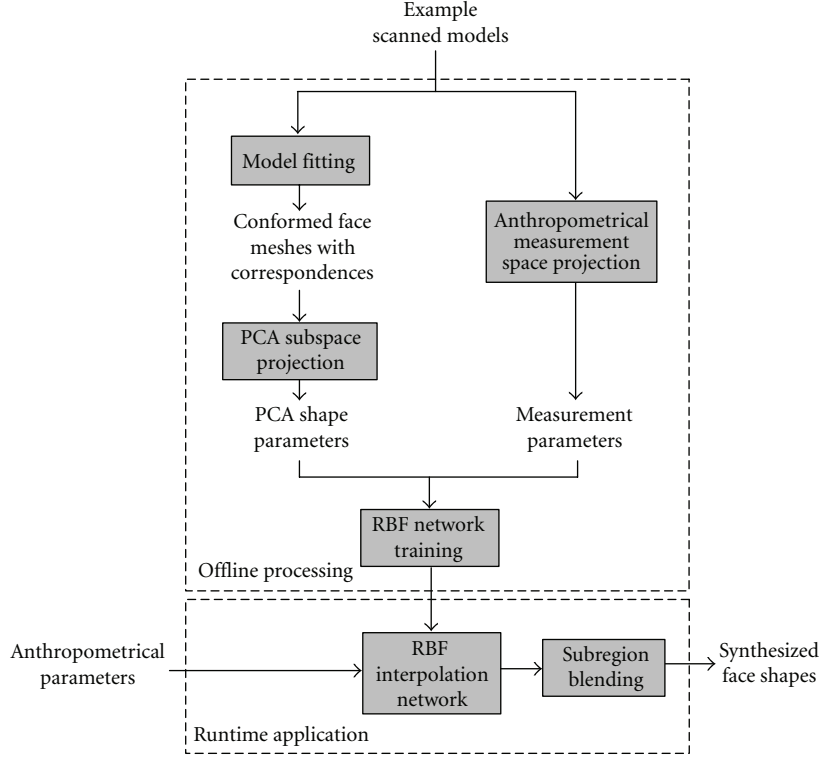


FIGURE 1: Overview of the interactive face shape synthesis system.

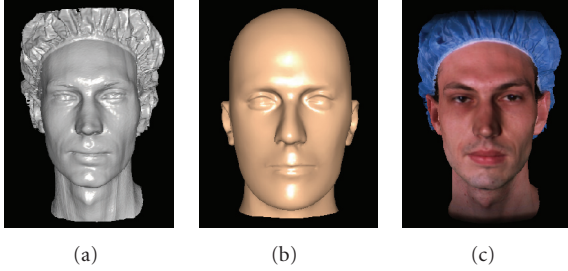


FIGURE 2: Face data: (a) scanned face geometry; (b) texture-mapped face scan; (c) generic model.

surface for all S_i using G under the constraint that \mathbf{v}_j is mapped to $\mathbf{u}_{i,j}$. The displacement vector $\mathbf{d}_{i,j} = \mathbf{u}_{i,j} - \mathbf{v}_j$ is known for each feature point \mathbf{v}_j on the generic model and $\mathbf{u}_{i,j}$ on the scanned surface. These displacements are utilized to construct the interpolating function that returns the displacement for each generic mesh vertex:

$$\mathbf{f}(\mathbf{x}) = \sum_{j=1}^n \mathbf{w}_j \phi_j(\|\mathbf{x} - \mathbf{v}_j\|) + \mathbf{M}\mathbf{x} + \mathbf{t}, \quad (1)$$

where $\mathbf{x} \in \mathcal{R}^3$ is a vertex on the generic model, $\|\cdot\|$ denotes the Euclidean norm and ϕ is a radial basis function. \mathbf{w}_j , \mathbf{M} and \mathbf{t} are the unknown parameters. Among them, $\mathbf{w}_j \in \mathcal{R}^3$ are the interpolation weights, $\mathbf{M} \in \mathcal{R}^{3 \times 3}$ represents rotation and scaling transformations, and $\mathbf{t} \in \mathcal{R}^3$ represents translation transformation.

Different functions for $\phi(r)$ are available [35]. We had better results with the multi-quadric function $\phi(r) = \sqrt{r^2 + \rho^2}$, where ρ is the locality parameter used to control how the basis function is influenced by neighboring feature points. ρ is determined as the Euclidean distance to the nearest other feature point. To determine the weights \mathbf{w}_j and the affine transformation parameters \mathbf{M} and \mathbf{t} , we solve the following equations:

$$\mathbf{d}_{i,j} = \mathbf{f}(\mathbf{v}_j)|_{j=1}^n, \quad \sum_{j=1}^n \mathbf{w}_j = 0, \quad \sum_{j=1}^n \mathbf{w}_j^T \mathbf{v}_j = 0. \quad (2)$$

This system of linear equations is solved using the LU decomposition to obtain the unknown parameters. Using the predefined interpolation function as given in (1), we calculate the displacement vectors of all vertices to deform the generic model.

3.2. Local Deformation. The warping with a small set of correspondences does not produce a perfect surface match. We further improve the shape using a local deformation which fits the globally warped generic mesh \tilde{G} to the scanned model S_i by iteratively minimizing the distance from the vertices of \tilde{G} to the surface of S_i . To optimize the positions of vertices of \tilde{G} , the local deformation process minimizes an energy function:

$$E(\tilde{G}) = E_{\text{ext}}(\tilde{G}, S_i) + E_{\text{int}}(\tilde{G}) \quad (3)$$

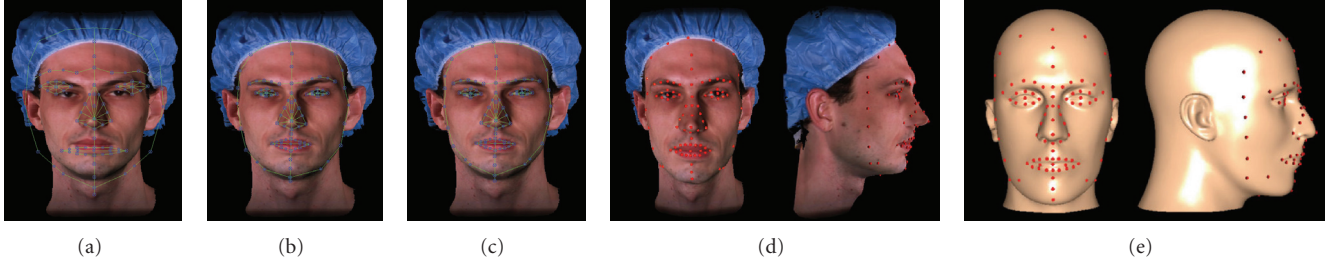


FIGURE 3: Semi-automatic feature point identification: (a) initial outline of the feature mask; (b) after automatic facial feature detection; (c) after interactive tuning; (d) and (e) 3D feature points identified on the scanned model and the generic model.

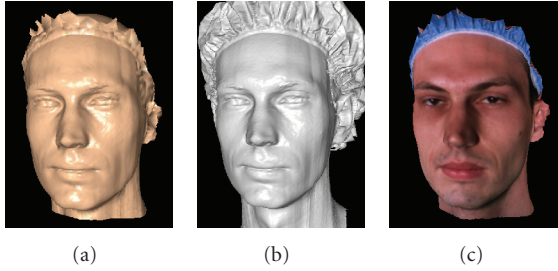


FIGURE 4: Model fitting: (a) deformed generic mesh after model fitting; (b) scanned model; (c) texture mapping of the deformed generic mesh.

where E_{ext} stands for the external energy and E_{int} the internal energy.

The external energy term E_{ext} attracts the vertices of \tilde{G} to their closest compatible points on S_i . It is defined as

$$E_{\text{ext}}(\tilde{G}, S_i) = \sum_{j=1}^{N_G} \zeta_j \|\mathbf{x}_j - \mathbf{s}_j\|^2, \quad (4)$$

where N_G is the number of vertices on the generic mesh, \mathbf{x}_j is the j th mesh vertex, and \mathbf{s}_j is the closest compatible point of \mathbf{x}_j on S_i . The weights ζ_j measure the compatibility of the points on \tilde{G} and S_i . As \tilde{G} closely approximates S_i in the global warping procedure, we consider a vertex on \tilde{G} and a point on S_i to be highly compatible if the surface normals at each point have similar directions. Hence, we define ζ_j as:

$$\zeta_j = \begin{cases} \mathbf{n}(\mathbf{x}_j) \cdot \mathbf{n}(\mathbf{s}_j) & \text{if } \mathbf{n}(\mathbf{x}_j) \cdot \mathbf{n}(\mathbf{s}_j) > 0 \\ 0 & \text{otherwise,} \end{cases} \quad (5)$$

where $\mathbf{n}(\mathbf{x}_j)$ and $\mathbf{n}(\mathbf{s}_j)$ are the surface normals at \mathbf{x}_j and \mathbf{s}_j , respectively. In this way, dissimilar local surface patches are less likely to be matched, for example, front-facing surfaces will not be matched to back-facing surfaces. To accelerate the minimum-distance calculation, we precompute a hierarchical bounding box structure for S_i so that the closest triangles are checked first.

The transformations applied to the vertices within a region of the surface may differ from each other considerably, resulting in a non-smoothly deformed surface. To enforce

local smoothness of the mesh, the internal energy term E_{int} is introduced as follows:

$$E_{\text{int}}(\tilde{G}) = \sum_{j=1}^{N_G} \sum_{k \in \Omega_j} \left\| (\mathbf{x}_j - \mathbf{x}_k) - (\tilde{\mathbf{x}}_j - \tilde{\mathbf{x}}_k) \right\|^2, \quad (6)$$

where Ω_j is the set grouping all neighboring vertices \mathbf{x}_k that are linked by edges to \mathbf{x}_j , and $\tilde{\mathbf{x}}_j$ and $\tilde{\mathbf{x}}_k$ are the original positions of \mathbf{x}_j and \mathbf{x}_k before local deformation. Including this energy term constrains the deformation of the generic mesh and keeps the optimization from converging to a solution far from the initial configuration.

Minimizing $E(\tilde{G})$ is a nonlinear least-square problem and optimization is performed using L-BFGS-B, which is a quasi-Newtonian solver [36]. The optimization stops when the difference between E at the previous and current iterations drops below a user-specified threshold. After the local deformation, each mesh vertex takes texture coordinates associated with its closest scanned data point for texture mapping. Finally, we reconstruct surface details in a hierarchical manner by taking advantage of the quaternary subdivision scheme and normal mesh representation [37]. Figure 4 shows the results of model fitting. Hence, a spatial correspondence is established by the generated normal meshes.

4. Forming Feature Shape Spaces

We perceive the face as a set of features. In this work, the global face shape is also regarded as a feature. Since all face scans are in correspondence through mapping onto the generic model, it is sufficient to define the feature regions on the generic model. We manually partition the generic model into four regions: eyes, nose, mouth and chin. This segmentation is transferred to all normal meshes to generate individualized feature shapes with correspondences (see Figure 5). In order to isolate the shape variation from the position variation, we normalize all scanned models with respect to the rotation and translation of the face before the model fitting process.

We form a shape space for each facial feature using PCA. Given the set $\Gamma = \{F\}$ of features, let $\{F_i\}_{i=1, \dots, M}$ be a set of example meshes of a feature F , each mesh being associated to one of the M scanned models in the database. These meshes are represented as vectors that contain the x , y , z

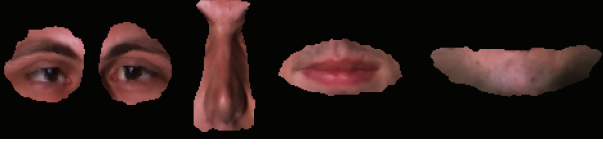


FIGURE 5: Four facial features decomposed from the level 2 normal mesh.

coordinates of N vertices $F_i = (x_1^i, y_1^i, z_1^i, \dots, x_N^i, y_N^i, z_N^i) \in \mathcal{R}^{3N}$. The average over M example meshes is given by $\psi_0 = (1/M) \sum_{i=1}^M F_i$. Each example mesh differs from the average by the vector $dF_i = F_i - \psi_0$. We arrange the deviation vectors into a matrix $\mathbf{C} = [dF_1, dF_2, \dots, dF_M] \in \mathcal{R}^{3N \times M}$. PCA of the matrix \mathbf{C} yields a set of M non-correlated eigenvectors ψ_i and their corresponding eigenvalues λ_i . The eigenvectors are sorted according to the decreasing order of their eigenvalues. Every example model can be regenerated using (7).

$$F_i(\alpha) = \psi_0 + \sum_{j=1}^K \alpha_{ij} \psi_j, \quad (7)$$

where $0 < K < M$ and $\alpha_{ij} = (F_i - \psi_0) \cdot \psi_j$ are the coordinates of the example mesh in terms of the reduced eigenvector basis. We choose the K such that $\sum_{i=1}^K \lambda_i \geq \tau \sum_{i=1}^M \lambda_i$, where τ defines the proportion of the total shape variation (98% in our experiments). In this model each eigenvector is a coordinate axis. We call these axes eigenmeshes.

5. Anthropometric Parameters

Face anthropometry provides a set of meaningful measurements or shape parameters that allow the most complete control over the shape of the face. Farkas [5] describes a widely used set of measurements to characterize the human face. The measurements are taken between the landmark points defined in terms of visually-identifiable or palpable features on the subject face using carefully specified procedures and measuring instruments. Such measurements use a total of 47 landmark points for describing the face. As described in Section 2, each example in our face scan database is equipped with 86 landmarks. Following the conventions laid out in [5], we have chosen a subset of 38 landmarks for anthropometric measurements (see Figure 6).

Farkas [5] describes a total of 132 measurements on the face and head. Instead of supporting all 132 measurements, we are only concerned with those related to five facial features (including global face outline). In this paper, 68 anthropometric measurements are chosen as shape control parameters. As an example, Table 1 lists the nasal measurements used in our work. The example models are placed in the standard posture for anthropometric measurements. In particular, the axial distances correspond to the x , y , and z axes of the world coordinate system. Such a systematic collection of anthropometric measurements is taken through all example models in the database to determine their locations in a multi-dimensional measurement space.

6. Feature Shape Synthesis

From the previous stage we obtain a set of examples of each facial feature with measured shape characteristics, each of them consisting of the same set of dimensions, where every dimension is an anthropometric measurement. The example measurements are normalized. Generally, we assume that an example model F_i of feature F has m dimensions, where each dimension is represented by a value in the interval $(0, 1]$. A value of 1 corresponds to the maximum measurement value of the dimension. The measurements of F_i can then be represented by the vector

$$\mathbf{q}_i = [q_{i1}, \dots, q_{im}], \quad \forall j \in [1, m] : q_{ij} \in (0, 1]. \quad (8)$$

This is equivalent to projecting each example model F_i into a *measurement space* spanned by the m selected anthropometric measurements. The location of F_i in this space is \mathbf{q}_i .

With the input shape control thus parameterized, our goal is to generate a new deformation of the facial feature by computing the corresponding eigenmesh coordinates with control through the measurement parameters. Given an arbitrary input measurement vector \mathbf{q} in the measurement space, such controlled deformation should interpolate the example models. To do this we interpolate the eigenmesh coordinates of the example models and obtain smooth range over the measurement space. Our feature shape synthesis problem is thus transformed to a scattered data interpolation problem. Again, the RBFs are employed. Given the input anthropometric control parameters, a novel output model with the desired shapes of facial features is obtained in runtime by blending the example models. Figure 7 illustrates this process. Our scheme first evaluates the predefined RBFs at the input measurement vector and then computes the eigenmesh coordinates by blending those of the example models with respect to the produced RBF values and pre-computed weight values. Finally, the output model with the desired feature shape is generated by evaluating the shape reconstruction model (7) at those eigenmesh coordinates. Note that there exist as many RBF-based interpolation functions as the number of eigenmeshes.

The interpolation is multi-dimensional. Consider a $\mathcal{R}^m \rightarrow \mathcal{R}$ mapping, the interpolated eigenmesh coordinates $a_j(\cdot) \in \mathcal{R}$, $1 \leq j \leq K$ at an input measurement vector $\mathbf{q} \in \mathcal{R}^m$ are computed as

$$a_j(\mathbf{q}) = \sum_{i=1}^M y_{ij} R_i(\mathbf{q}) \quad \text{for } 1 \leq j \leq K, \quad (9)$$

where $y_{ij} \in \mathcal{R}$ are the radial coefficients and M is the number of example models. Let \mathbf{q}_i ($1 \leq i \leq M$) be the measurement vector of an example model. The radial basis function $R_i(\mathbf{q})$ is a multi-quadric function of the Euclidean distance between \mathbf{q} and \mathbf{q}_i in the measurement space:

$$R_i(\mathbf{q}) = \sqrt{\|\mathbf{q} - \mathbf{q}_i\|^2 + \rho_i^2} \quad \text{for } 1 \leq i \leq M, \quad (10)$$

where ρ_i is the locality parameter used to control the behavior of the basis function and determined as the

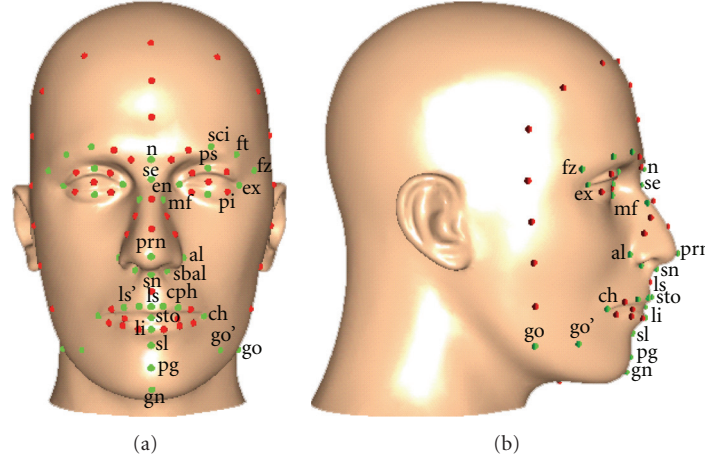


FIGURE 6: Head geometry with anthropometric landmarks (green dots). The landmark names are taken from [5].

TABLE 1: Anthropometric measurements of the nose.

Landmarks	Measurement Name	Landmarks	Measurement Name
mf-mf	Nasal root width	n-pm	Nasal bridge length
al-al	Nose width	al-pm	Ala surface length
sb-al-sb	Alar base width	al-sn	Alar point-subnasale length
sb-al-sn	Nostril floor width	n-pm	Inclination of the nasal bridge
sn-pm	Nasal tip protrusion	sn-prn	Inclination of the columella
en-se	Nasal root depth	al-pm	Inclination of the alar-slope line
en-se	Nasal root slope	n-se-pm	Nasofrontal angle
al-pm	Ala length	al-pm-al	Ala-slope angle
al-mf	Nasal bridge angle	se-pm-sn	Nasal tip angle
n-sn	Nose height	pm-sn-ls	Nasolabial angle

Euclidean distance between \mathbf{q} and the closest other example measurement vector.

The j th eigenmesh coordinate of the i th example model, a_{ij} , corresponds to the measurement vector of the i th example model, \mathbf{q}_i . Equation (9) should be satisfied for \mathbf{q}_i and a_{ij} ($1 \leq i \leq M$):

$$a_{ij} = \sum_{i=1}^M \gamma_{ij} R_i(\mathbf{q}_i) \quad \text{for } 1 \leq j \leq K. \quad (11)$$

The radial coefficients γ_{ij} are obtained by solving this linear system using an LU decomposition. We can then generate the eigenmesh coordinates, hence the shape, corresponding to the input measurement vector \mathbf{q} according to (9).

7. Subregion Shape Blending

After the shape interpolation procedure, the surrounding facial areas should be blended with the deformed internal facial features to generate a seamlessly smooth face mesh. The position of a vertex \mathbf{x}_i in the feature region F after deformation is \mathbf{x}'_i . Let \mathcal{V} denote the set of vertices of the head mesh. For smooth blending, positions of the subset $\overline{\mathcal{V}}_F = \mathcal{V} \setminus \mathcal{V}_F$ of vertices of \mathcal{V} that are not inside the feature region should be updated with deformation of the

facial features. For each vertex $\mathbf{x}_j \in \overline{\mathcal{V}}_F$, the vertex in each feature region that exerts influence on it, $\mathbf{x}_{k_i}^F$, is the one of minimal distance to it. It is desirable to use geodesic distance on the surface, rather than Euclidean distance to measure the relative positions of two mesh vertices. We adopt an approximation of the geodesic distance based on a cylindrical projection which is preferable for regions corresponding to a volumetric surface (e.g., the head). The idea is that distance between two vertices on the projected mesh in the 2D image plane is a fair approximation of geodesic distance. Thus, $\mathbf{x}_{k_i}^F$ is obtained as:

$$\|\mathbf{x}_j - \mathbf{x}_{k_i}^F\|_G \approx \min_{\{i|i \in \mathcal{V}_F\}} \|\mathbf{x}_j^* - \mathbf{x}_i^*\|, \quad (12)$$

where \mathbf{x}_i^* and \mathbf{x}_j^* are the positions of vertices on the projected mesh, and $\|\cdot\|_G$ denotes the geodesic distance. Note that the distance is measured offline in the original undeformed generic mesh. For each non-feature vertex \mathbf{x}_j , its position is updated in shape blending as:

$$\mathbf{x}'_j = \mathbf{x}_j + \sum_{F \in \Gamma} \exp\left(-\frac{1}{\alpha} \|\mathbf{x}_j - \mathbf{x}_{k_i}^F\|_G\right) \|\mathbf{x}_{k_i}^F - \mathbf{x}_{k_i}^F\|, \quad (13)$$

where Γ is the set of facial features and α controls the size of the region influenced by the blending. We set α to 1/10 of the diagonal length of the bounding box of the head model.

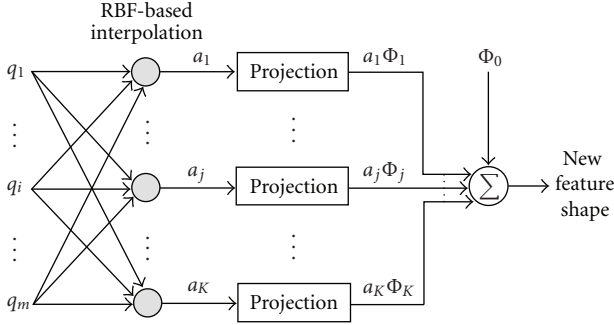


FIGURE 7: Generating a new facial feature shape by blending example models through interpolation of their eigenmesh coordinates.

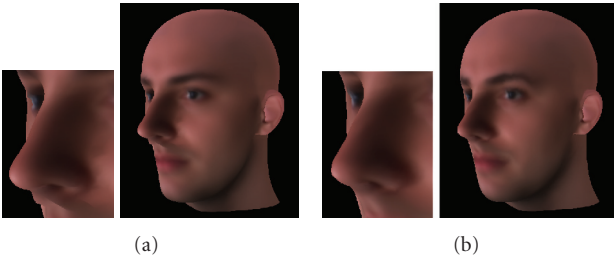


FIGURE 8: Synthesis of the nose shape: (a) Without shape blending, the obvious geometric discontinuities around the boundary of the nose region impair realism of the synthesis to a large extent. (b) Using our approach, the geometries of the feature region and surrounding areas are smoothly blended around their boundary.

Figure 8(b) shows the effect of our shape blending scheme employed in synthesizing the nose shape.

8. Results

Our method has been implemented in an interactive system with C++/OpenGL, where the user can select facial features to work on interactively. A GUI snapshot is shown in Figure 9. Our system starts with a mean model which is computed as the average of 186 meshes of the RBF-warped models and textured with the mean cylindrical full-head texture image [38]. Our system also allows the user to select the desired feature(s) from a database of pre-constructed typical features, which are shown in the small icons on the upper-left of the GUI. Upon selecting a feature from the database, the feature will be imported seamlessly into the displayed head model and can be further edited if needed. The slider positions for each of the available feature in the database are stored by the system so that their configuration can be restored whenever the feature is chosen. Such a feature importing mode enables coarse-to-fine modification of features, making the face synthesis process less tedious. We invited several student users who naturally lack the graphics professional's modeling background to create face models using our system. The laymen appreciated the intuitiveness and continuous variability of the control sliders. Table 2 shows the details of the datasets.

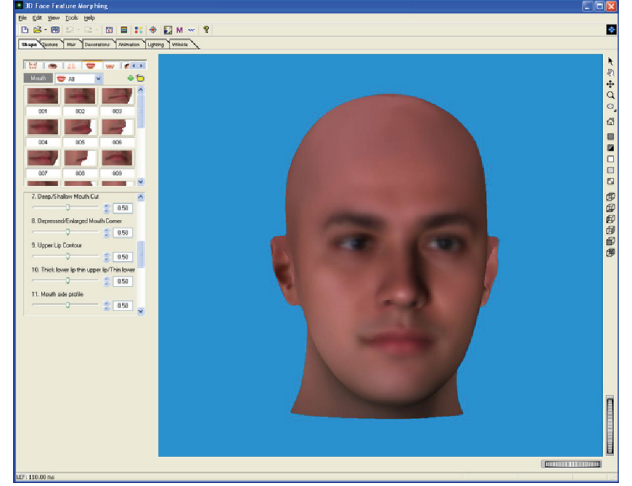


FIGURE 9: GUI of our system.

TABLE 2: Details of the data used in our system. M is the number of examples, N is the number of mesh vertices (the number of original dimensions equals $3N$), K is the number of reduced dimensions of the PCA space, and m is the number of anthropometric control parameters.

	Full head	Eyes	Nose	Mouth	Chin
M	186	186	186	186	186
N	16192	2914	1782	2105	643
K	34	23	26	20	18
m	16	13	20	12	7

Figure 10 illustrates a number of distinct facial shapes synthesized to satisfy user-specified local shape constraints. Clear differences are found in the width of the nose alar wings, the straightness of the nose bridge, the inclination of the nose tip, the roundness of eyes, the distance between eyebrows and eyes, the thickness of mouth lips, the shape of the lip line, the sharpness of the chin, and so forth. A morphing can be generated by varying the shape parameters continuously, as shown in Figures 10(b) and 10(c). In addition to starting with the mean model, the user may also select the desired head model of a specific person from the example database for further editing. Figure 11 illustrates face editing results on the models of two individuals for various user-intended characteristics.

In order to quantify the performance, we arbitrarily selected ten examples in the database for the cross validation. Each example has been excluded from the example database in training the face synthesis system and its shape measurements were used as a test input to the system. The output model was then compared against the original model. Figure 12 shows a visual comparison of the result. We assess the reconstruction by measuring the maximum, mean, and root mean square (RMS) errors from the feature regions of the output model to those of the input model. The 3D errors are computed by the Euclidean distance between each vertex of the ground truth and synthesized model. Table 3 shows the average errors measured for the ten reconstructed models.

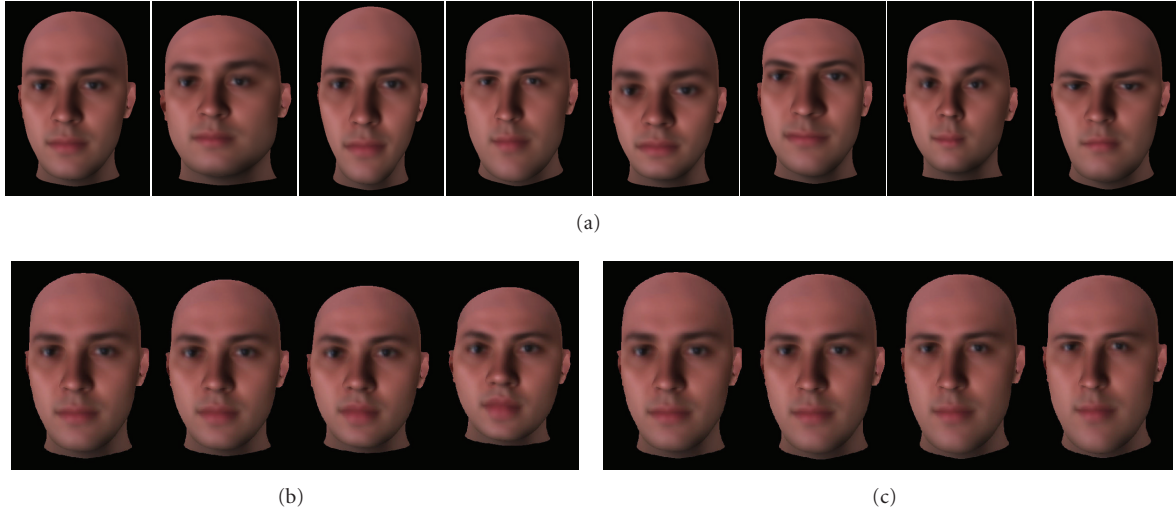


FIGURE 10: (a) New faces synthesized from the average model (leftmost) with global and local shape variations. (b) and (c) Face shape morphing (left to right in each example).

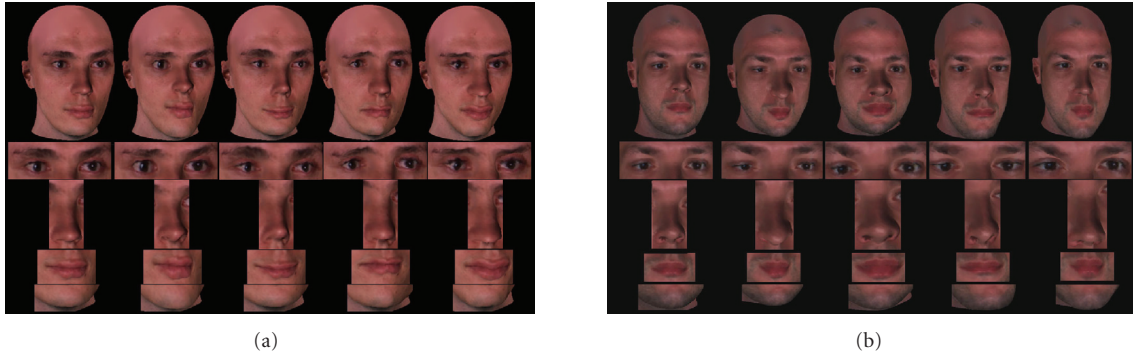


FIGURE 11: Feature-based face editing on the models of two individuals. In each example, the original model is shown in the top-left.

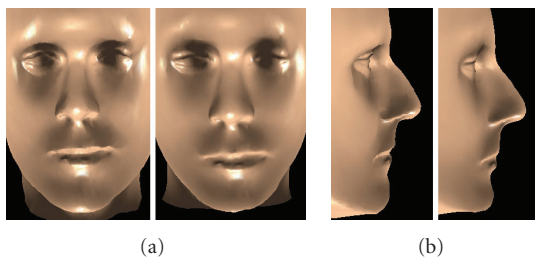


FIGURE 12: Comparison of an original model (left in each view) and synthesized model (right in each view) in cross validation.

The errors are given using both absolute measures (/mm) and as a percentage of the diameter of the output head model bounding box.

We compare our method against the approach of optimization in the PCA space (Opt-PCA). Opt-PCA performs optimization to estimate weights of the eigen-model (7). It starts from the mean model on which the anthropometric landmarks are in their source positions. The corresponding target positions of these landmarks are the landmark positions on the example model. We then optimize the mesh

shape in the subspaces of facial features using the downhill simplex algorithm such that the sum of distances between the source and target positions of all landmarks is minimized. Table 4 shows the comparison between our method and Opt-PCA. Opt-PCA produces a large error since the number of landmarks is small and it is not sufficient to fully determine weights of the eigen-model. Opt-PCA is also slow since there are many PCA weights to be optimized iteratively.

Our system runs on a 2.8 GHz PC with 1 GB of RAM. Table 5 shows the time cost of different procedures. At run-time, our scheme spends less than one second in generating a new face shape upon receiving the input parameters. This includes the time for the evaluation of RBF-based interpolation functions and for shape blending around the feature region boundaries.

9. Applications

Apart from creating plausible 3D face models from users' descriptions, our feature-based face reconstruction approach is useful for a range of other applications. The statistics of facial features allow analysis of their shapes, for instance,

TABLE 3: Cross validation results of our 3D face synthesis system.

	Eyes	Nose	Mouth	Chin
Average max.	3.85 (0.91%)	2.55 (0.84%)	2.86 (0.94%)	4.46 (1.06%)
Average mean	2.57 (0.57%)	1.62 (0.38%)	2.04 (0.49%)	2.25 (0.53%)
Average RMS	3.62 (0.86%)	2.23 (0.53%)	2.84 (0.67%)	3.14 (0.74%)

TABLE 4: Comparison of our method with the optimization approach. Each value is an average of ten trials with different example models.

	Opt_PCA				Our method			
	Eyes	Nose	Mouth	Chin	Eyes	Nose	Mouth	Chin
Mean error (mm)	2.83	3.27	3.84	6.65	2.57	1.62	2.04	2.25
Time (s)	34.8	21.5	23.5	5.3	0.4	0.5	0.4	0.3

TABLE 5: Time consumed for different processes of system implementation. For some processes (in *italic*), the time spent per example is shown. Notation: time consumed in interactive operation (T_I), time consumed in automatic computation (T_A).

Process	T_I	T_A
Offline processing		
<i>Feature point identification</i>	3–5 minutes	6 seconds
<i>Global warping</i>	N/A	2 seconds
<i>Local deformation</i>	N/A	4 minutes
<i>Multi-resolution model generation</i>	N/A	5 seconds
Computing eigenmeshes by PCA	N/A	2 hours
<i>Computing eigenmesh coordinates</i>	N/A	0.5 seconds
<i>Computing anthropometric measurements</i>	N/A	0.2 seconds
LU decomposition	N/A	2 minutes
Runtime		
Feature shape synthesis	N/A	0.6 seconds

to discern differences between groups of faces. They also allow synthesis of new faces for applications such as facial feature transfer between different faces and adaptation of the model to local populations. Moreover, our approach allows for compression of 3D face data, facilitating us to share statistics with other researchers to allow the synthesis and further study of high-resolution faces.

9.1. Analyzing the Shape of Facial Features. As the first application, we consider analysis of the shape of facial features. This is useful for classification of face scans. We wish to gain insight into how facial features change with personal characteristics by comparing statistics between groups of faces. We calculate the mean and standard deviation statistics of anthropometric measurements for each facial feature of different groups. The morphometric differences between groups are visualized by comparing the statistics of each facial feature in a diagram. We follow this approach to study the effects of race and gender.

Race. To investigate how the shape of facial features changes with race, we compare three groups of 18–30 year-old Caucasian (72 subjects), Mongolian (18 subjects), and Negroid (26 subjects) which are divided almost equally between the

genders. The group statistics are shown in Figure 13, colored with blue, green, and red, respectively. It shows that the Caucasian nose is narrow, the Mongolian nose is medial, and the Negroid nose is wide. The statistics indicate a relatively protruding, narrow nose in Caucasian. The Mongolian nose is less protruding and wider, and the Negroid nose has the smallest protrusion. The nasal root depth and nasofrontal angle are the largest for the Caucasian, exhibiting significant differences compared with the smaller Negroid and smallest Mongolian values. This suggests the high nasal root in Caucasian and relatively flat nasal root in Negroid and Mongolian. Significant differences among the three races are also found in inclination of the columella and nasal tip angle, indicating the hooked nose in Caucasian and the snub nose in Mongolian and Negroid.

For the eyes, the main characteristics of the Caucasian group are the largest eye fissure height, the smallest intercanthal width and eye fissure inclination angle. These suggest that the Caucasian eyes typically have larger openings with horizontally aligned inner and external eye corners. The Mongolian group has the largest intercanthal width, and the greatest inclination in the shortest eye fissure and the smallest eye fissure height, which indicate the relatively small eye openings separated in a large horizontal distance with positions of the inner eye corners lower than those of the external ones. Blacks have the largest eye fissure length and binocular with, which denote the relatively wide eyes in this group.

As shown in Figure 13(c), many measurements of the mouth of Negroid (e.g., mouth width, upper and lower lip height, upper and lower vermilion height) are the largest among the three races. They are significantly different from those in Caucasian or Mongolian group. Mongolian has the relatively narrow mouth and thin lips. In Caucasian the skin portion of the upper and lower lips and their vermilion height are the smallest. However, the proportions of the upper and lower lip heights in the three races reveal the similarity.

From statistics illustrated in Figure 13(d), the Negroid chin has the characteristics of a long vertical profile dimension and small width. The smallest value of inclination of the chin from the vertical and the largest mentocervical angle also indicates a less protruding chin for Negroid. In

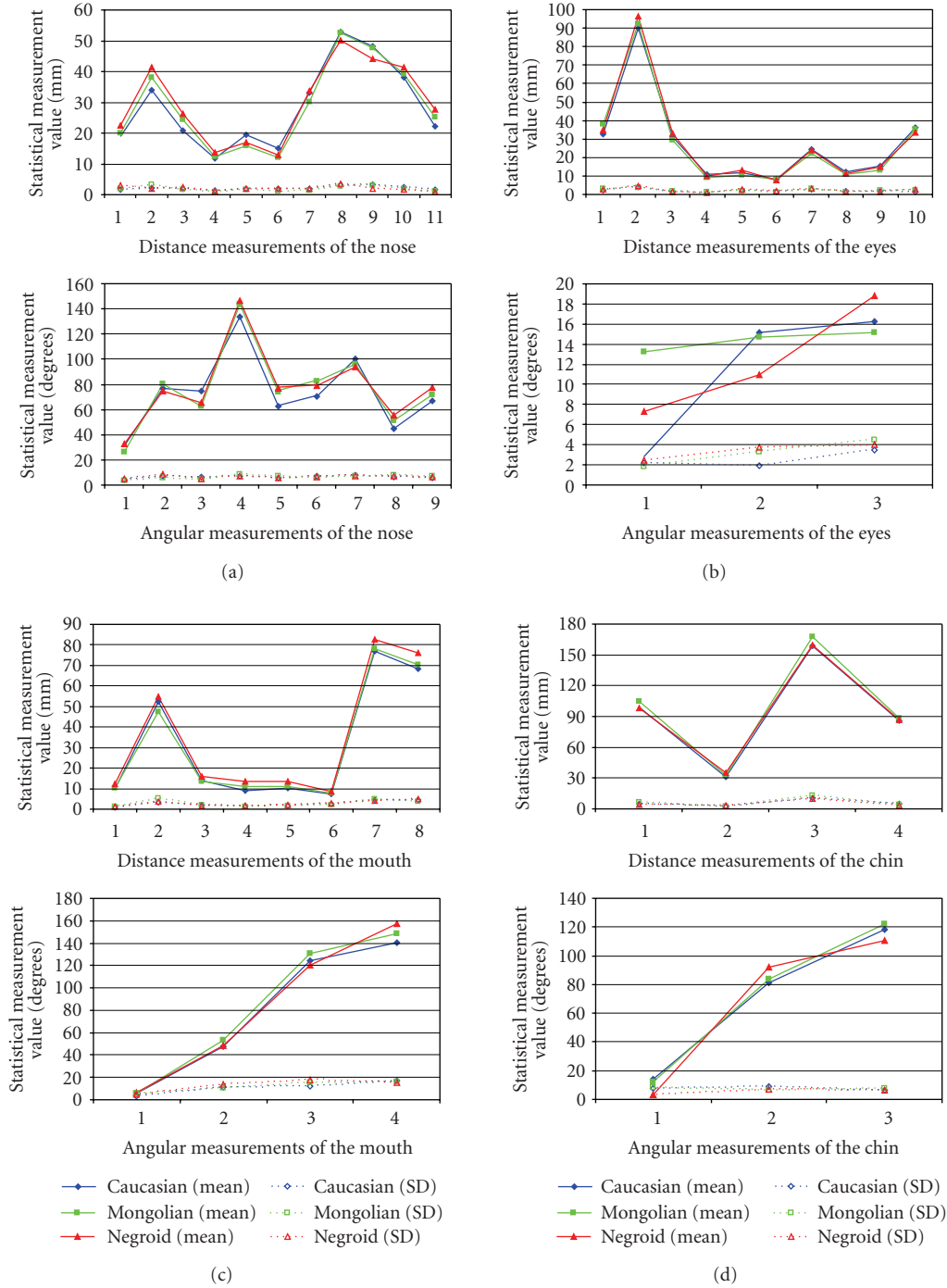


FIGURE 13: Comparison of statistics of facial feature measurements between races (blue, green and red for groups of Caucasian, Mongolian and Negroid, resp.). Each facial feature: statistics of the distance measurements (top) and statistics of the angular measurements (bottom).

Mongolian, the chin is the widest among the three races. The smallest chin height is found in Caucasian. Also, the chin of Caucasian is slightly wider than that of Negroid, but markedly narrower than that of Mongolian.

Gender. To study the effect of gender, we compare in Figure 14 18–30-year-old Caucasian females (35 subjects, in red) to Caucasian males of the same age group (37 subjects,

in blue). The change of the shape of facial features from females to males is different in character from that of the change between varying racial groups. The larger values of most distance measurements of the nose indicate that males have wide alar wings and wide, long nose bridge. The value of the nasal root depth is also indicative of high upper nose bridge of the male subjects. In females, the nose bridge and alar are narrower; the nose tip is sharper and

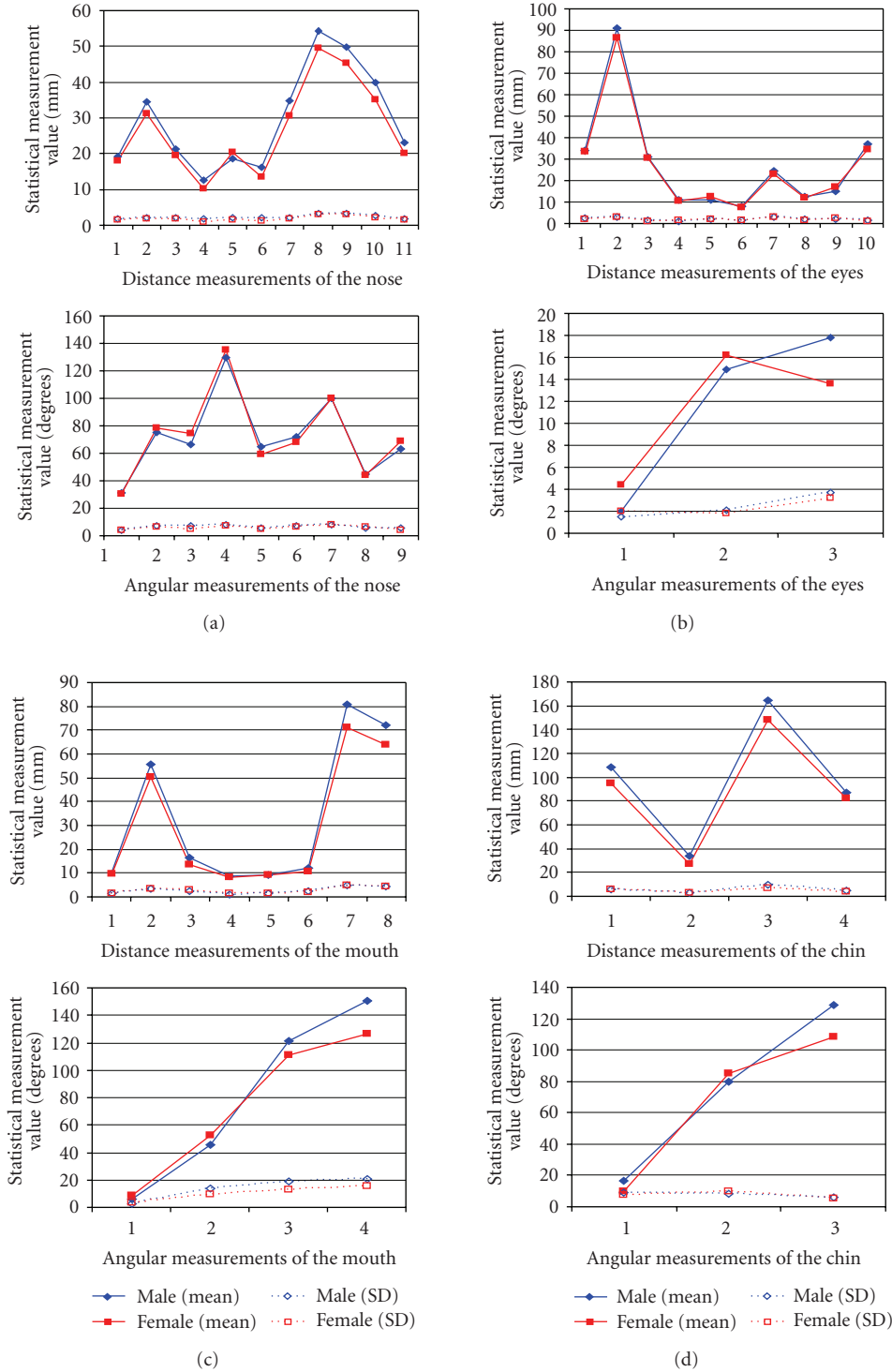


FIGURE 14: Comparison of statistics of facial feature measurements between genders (females in red and males in blue). Each facial feature: statistics of the distance measurements (top) and statistics of the angular measurements (bottom).

more protruding. In addition, the vertical profile around the junction of the nose bridge and the anterior surface of the forehead in females is flatter, which is suggested by the larger nasofrontal angle. The inclination of the nose bridge and columella reveals the similarity in two genders.

Regarding anthropometric measurements of the eyes, males have the larger intercanthal width and binocular width, which imply that their eyes are more separated with regard to the sagittal plane (vertical plane cutting through the center of the face). The width of the eye fissure of males

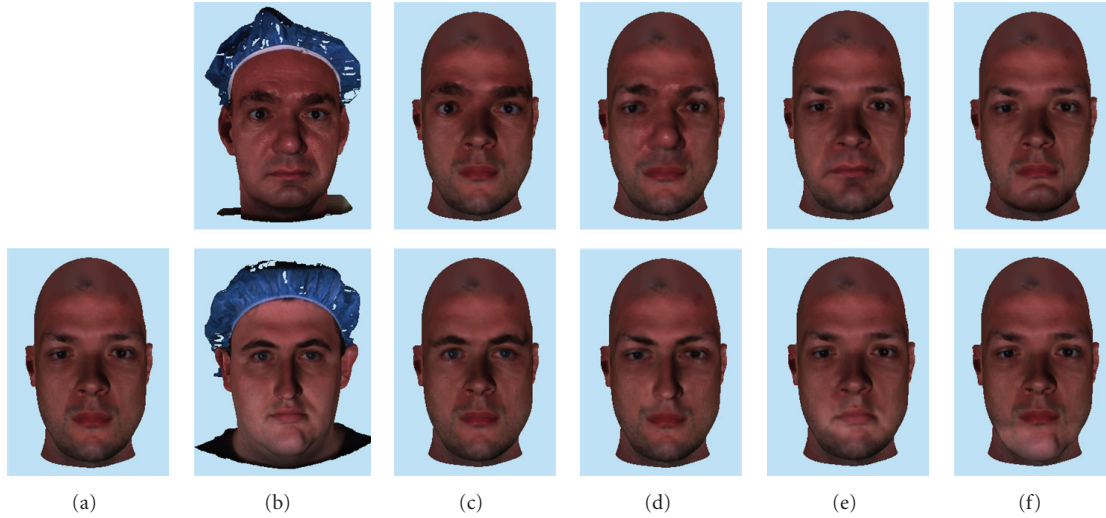


FIGURE 15: Transfer of facial features. We start with a source model (a) and synthesize facial features of the eyes (c), nose (d), mouth (e) and chin (f) on it by coercing the shape parameters to match those of two example faces (b).

is slightly larger than that of females, whereas the heights of the eye fissure of two genders are similar. Males also have the large height of the lower eyelid. In females, the height of the upper eyelid and distance between eyebrows and eyes are larger. Another characteristic of females is the large inclination of the eye fissure.

Most distance measurements of the mouth in the male group are larger in both genders, as shown in Figure 14(c). This suggests that males have a much wider mouth with the large skin portion of the upper and lower lips. However, the vermilion heights of the upper and lower lips in two groups reveal the similar thickness of the lips in two genders. The differences exhibited in the angular measurements are indicative of more protruding lips and convex lip line of the female subjects.

The diagram in Figure 14(d) shows that the chin of males is characterized by large size in three dimensions (width, height and depth) due to the large underlying mandible. The greater inclination angle of the chin and smaller mentocervical angle also indicate a relatively protruding chin in males compared to that of females.

9.2. Facial Feature Transfer. In the applications of creating virtual characters for entertainment production, sometimes it is desirable to adjust the face so that it has certain facial features similar to those of a particular person. Therefore, it is useful to be able to transfer desired facial feature(s) between different human subjects. One might wish, given a database of example faces, to select a face or multiple faces to which to adjust facial features.

Our high-level facial feature control framework allows the transfer of desired facial features from example faces to a source model in a straightforward manner. We can alter the feature of the source model with a feature-adjustment step which coerces the anthropometric measurement vector to match that of the target feature of an example face. The new shape of the selected feature is reconstructed on the source model and can be further edited if needed.

Figure 15(a) shows the source model which is the approximation of an example 3D scan using the deformed generic mesh. Figures 15(c) to 15(f) show the results of matching the shape measurements of the features of this model to those of two example faces shown in Figure 15(b). The synthesis keeps global shape of the source model, while transferring features of the target subject to the source subject. With decomposition of the face into local features, typical features of different target faces can be transferred in conjunction with each other to the same source model. Figure 16 shows a composite face built from facial features of four individuals.

9.3. Face Adaptation to Local Populations. Adapting the model to local populations falls neatly into our framework. The problem of automatically generating a population is reduced to the problem of generating the desired number of plausible sets of control parameters. It is convenient to generate each parameter value independently as if sampled from the Gaussian normal distribution with its mean and variance. The generated control parameter values both respect a given population distribution, and—thanks to the use of interpolation in the local feature shape spaces—produce a believable face. The examples of this process are shown in Figure 17.

9.4. Face Data Compression and Dissemination. For the face synthesis based on a large example data set, the ability to organize examples into database, compress, and efficiently transmit them is a critical issue. The example face meshes used for this paper are restricted from being transmitted in their full resolution because of their dense-data nature. In our method, we take advantage of the fact that the objects under our consideration are of the same class and that they lie in correspondence to compress data very efficiently. Instead of storing instances of geometry data for every example, we adopt a compact representation obtained by extracting the statistics with PCA, which are several orders of magnitude smaller than the original 3D scans. This accounts

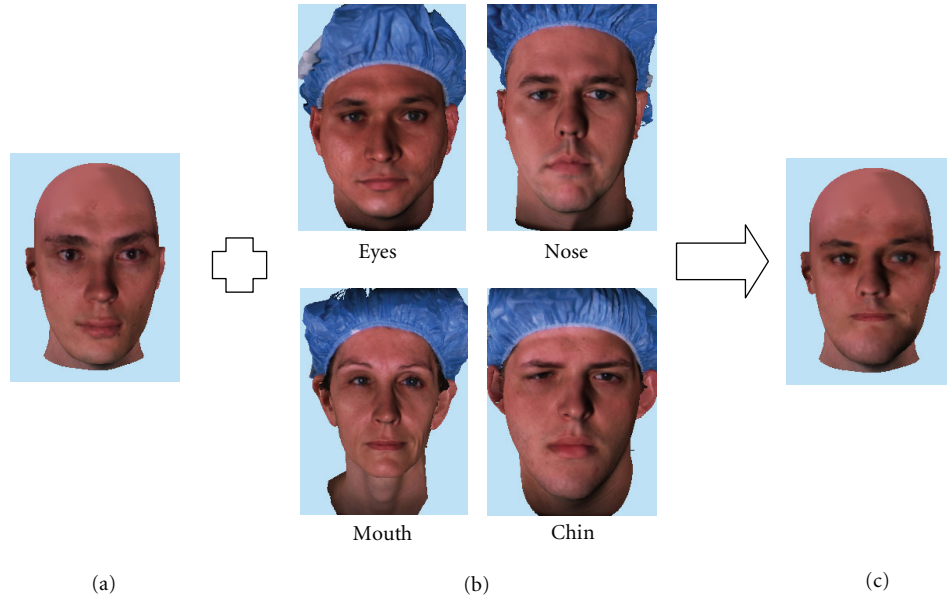


FIGURE 16: Facial features of four example faces (b) in our database are transferred to the source model (a) to generate a novel composite face (c).

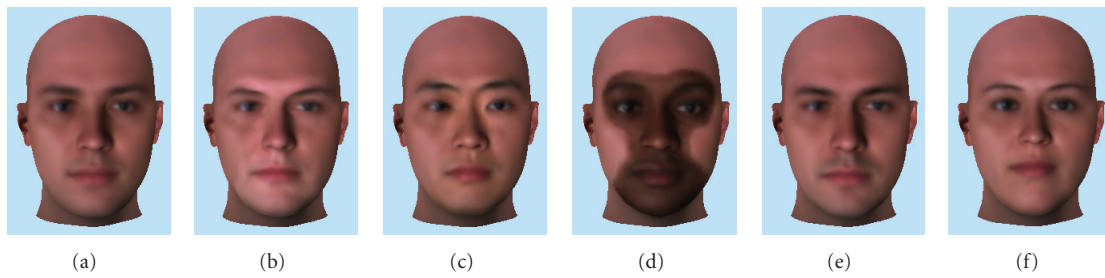


FIGURE 17: Adapting the face to population groups: (a) average face; (b), (c) and (d) synthesized faces with the ethnicity of Caucasian, Mongolian and Negroid, respectively; (e) and (f) synthesized male and female faces, respectively.

for the space gain from M times the dimensionality of high-resolution 3D scans (hundreds of thousands), to K ($K \leq M$) times the dimensionality of an eigenmesh (several thousands), with M and K being the number of examples and eigenmeshes respectively. For all faces, we also make available the statistics of facial feature measurements within different population groups. These statistics along with the eigenmeshes should make it possible for other researchers to investigate new applications beyond the ones described in this paper.

10. Conclusion and Future Work

We have presented an automatic runtime system for generating varied, realistic face models. The system automatically learns a statistical model from example meshes of facial features and enforces it as a prior to generate/edit the face model. We parameterize the feature shape examples using a set of anthropometric measurements, projecting them into the measurement spaces. Solving the scattered data

interpolation problem in a reduced subspace yields a natural face shape that achieves the goals specified by the user. With an intuitive slider interface, our system appeals to both beginning and professional users, and greatly reduces the time for creating natural face models compared to existing 3D mesh editing software. With the anthropometrics-based face synthesis, we explore a variety of applications, including analysis of facial features in subjects with different races, transfer of facial features between individuals, and adjusting the apparent race and gender of faces.

The quality of the generated model depends on the model priors. Therefore, an appropriate database with large number and variety of the faces must be available. We would like to extend our current database to incorporate more 3D face examples of Mongolian and Negroid races as well as to increase the diversity of age. We also plan to increase the number of facial features to choose from. To improve the system interface, we would like to integrate the “dragging” interaction mode which allows for directly choosing one or more feature points of a facial feature and then dragging

them to the desired positions to generate a new facial shape. This involves updating multiple anthropometric parameters in one step and results in large scale changes.

References

- [1] "Autodesk Maya," <http://www.autodesk.com/maya>.
- [2] "Poser 7," <http://graphics.smithmicro.com/go/poser>.
- [3] "DazStudio," <http://www.daz3d.com>.
- [4] "PeoplePutty," <http://www.haptek.com>.
- [5] L. G. Farkas, *Anthropometry of the Head and Face*, Raven Press, New York, NY, USA, 1994.
- [6] F. I. Parke and K. Waters, *Computer Facial Animation*, AK Peters, Wellesley, Mass, USA, 1996.
- [7] J. Y. Noh and U. Neumann, "A survey of facial modeling and animation techniques," USC Technical Report 99-705, Univeristy of Southern Californina, Los Angeles, Calif, USA, 1999.
- [8] S. DiPaola, "Extending the range of facial types," *Journal of Visualization and Computer Animation*, vol. 2, no. 4, pp. 129–131, 1991.
- [9] N. Magnenat-Thalmann, H. T. Minh, M. de Angelis, and D. Thalmann, "Design, transformation and animation of human faces," *The Visual Computer*, vol. 5, no. 1-2, pp. 32–39, 1989.
- [10] F. I. Parke, "Parameterized models for facial animation," *IEEE Computer Graphics and Applications*, vol. 2, no. 9, pp. 61–68, 1982.
- [11] M. Patel and P. Willis, "Faces: the facial animation, construction and editing system," in *Proceedings of the European Computer Graphics Conference and Exhibition (Eurographics '91)*, pp. 33–45, Vienna, Austria, September 1991.
- [12] T. Akimoto, Y. Suenaga, and R. S. Wallace, "Automatic creation of 3D facial models," *IEEE Computer Graphics and Application*, vol. 13, no. 5, pp. 16–22, 1993.
- [13] B. Guenter, C. Grimm, D. Wood, H. Malvar, and F. Pighin, "Making faces," in *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '98)*, pp. 55–65, Orlando, Fla, USA, July 1998.
- [14] C. J. Kuo, R.-S. Huang, and T.-G. Lin, "3-D facial model estimation from single front-view facial image," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 3, pp. 183–192, 2002.
- [15] W.-S. Lee and N. Magnenat-Thalmann, "Fast head modeling for animation," *Image and Vision Computing*, vol. 18, no. 4, pp. 355–364, 2000.
- [16] Z. Liu, Z. Zhang, C. Jacobs, and M. Cohen, "Rapid modeling of animated faces from video," *Journal of Visualization and Computer Animation*, vol. 12, no. 4, pp. 227–240, 2001.
- [17] I. K. Park, H. Zhang, V. Vezhnevets, and H. K. Choh, "Image-based photorealistic 3d face modeling," in *Proceedings of the 6th IEEE International Conference on Automatic Face and Gesture Recognition (FGR '04)*, pp. 49–54, Seoul, Korea, May 2004.
- [18] F. Pighin, J. Hecker, D. Lischinski, R. Szeliski, and D. H. Salesin, "Synthesizing realistic facial expressions from photographs," in *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '98)*, pp. 75–84, Orlando, Fla, USA, July 1998.
- [19] R. Enciso, J. Li, D. Fidaleo, T.-Y. Kim, J.-Y. Noh, and U. Neumann, "Synthesis of 3d faces," in *Proceedings of the 1st USF International Workshop on Digital and Computational Video (DCV '99)*, pp. 8–15, Tampa, Fla, USA, December 1999.
- [20] K. Kähler, J. Haber, H. Yamauchi, and H.-P. Seidel, "Head shop: generating animated head models with anatomical structure," in *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pp. 55–63, San Antonio, Tex, USA, July 2002.
- [21] K. Kähler, J. Haber, and H.-P. Seidel, "Geometry-based muscle modeling for facial animation," in *Proceedings of Graphics Interface*, pp. 37–46, Ottawa, Canada, June 2001.
- [22] Y. Lee, D. Terzopoulos, and K. Waters, "Realistic modeling for facial animation," in *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '95)*, pp. 55–62, Los Angeles, Calif, USA, August 1995.
- [23] D. DeCarlo, D. Metaxas, and M. Stone, "An anthropometric face model using variational techniques," in *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '98)*, pp. 67–74, Orlando, Fla, USA, July 1998.
- [24] V. Blanz and T. Vetter, "A morphable model for the synthesis of 3d faces," in *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '99)*, pp. 187–194, Los Angeles, Calif, USA, August 1999.
- [25] V. Blanz, I. Albrecht, J. Haber, and H.-P. Seidel, "Creating face models from vague mental images," *Computer Graphics Forum*, vol. 25, no. 3, pp. 645–654, 2006.
- [26] D. Vlasic, M. Brand, H. Pfister, and J. Popović, "Face transfer with multilinear models," in *Proceedings of the 32nd International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '05)*, pp. 426–433, Los Angeles, Calif, USA, July-August 2005.
- [27] T.-P. G. Chen and S. Fels, "Exploring gradient-based face navigation interfaces," in *Proceedings of Graphics Interface*, pp. 65–72, London, Canada, May 2004.
- [28] "PROfitTM from ABM United Kingdom Ltd.," <http://www.abm-uk.com>.
- [29] "E-FITTM from Aspley Ltd.," <http://www.efit.co.uk>.
- [30] "Identi-Kit.NETTM from Smith & Wesson®," <http://www.identikit.net>.
- [31] "FaceGen Modeller 3.0 from Singular Inversions Inc.," <http://www.FaceGen.com>.
- [32] "USF DARPA HumanID 3D Face Database," Courtesy of Prof. Sudeep Sarkar, University of South Florida, Tampa, Fla, USA.
- [33] ISO/IEC, "Overview of the MPEG-4 standard," <http://www.chiariglione.org/mpeg/standards/mpeg-4/mpeg-4.htm>.
- [34] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham, "Active shape models-their training and application," *Computer Vision and Image Understanding*, vol. 61, no. 1, pp. 38–59, 1995.
- [35] J. C. Carr, R. K. Beatson, J. B. Cherrie, et al., "Reconstruction and representation of 3D objects with radial basis functions," in *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '01)*, pp. 67–76, Los Angeles, Calif, USA, August 2001.
- [36] C. Zhu, R. H. Byrd, P. Lu, and J. Nocedal, "Algorithm 778: L-BFGS-B: fortran subroutines for large-scale bound-constrained optimization," *ACM Transactions on Mathematical Software*, vol. 23, no. 4, pp. 550–560, 1997.
- [37] I. Guskov, K. Vidimčev, W. Sweldens, and P. Schroöder, "Normal meshes," in *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '00)*, pp. 95–102, New Orleans, La, USA, July 2000.
- [38] Y. Zhang, "An efficient texture generation technique for human head cloning and morphing," in *Proceedings of the International Conference on Computer Graphics Theory and Applications (GRAPP '06)*, pp. 267–278, Setúbal, Portugal, February 2006.